Introducción

En las redes de computadoras, cuando dos dispositivos necesitan comunicarse de forma directa y confiable, se utilizan protocolos de la capa de enlace de datos del modelo OSI.

Entre los protocolos diseñados para enlaces punto a punto destacan SLIP, HDLC y PPP:

- SLIP: el más antiguo y básico, usado principalmente para conexiones seriales.
- HDLC: más seguro y confiable, utilizado especialmente entre routers.
- PPP: el más flexible y completo, compatible con autenticación, compresión y múltiples protocolos de red.

¿Qué son y para qué sirven?

SLIP, HDLC y PPP son protocolos fundamentales de la **capa de Enlace de Datos** porque permiten la transmisión ordenada y confiable de datos punto a punto a través de enlaces directos como cables seriales, líneas dedicadas o conexiones módem. Su importancia radica en que garantizan **compatibilidad, control de errores, autenticación y estructuración adecuada de la información**, asegurando una comunicación eficiente y segura entre dispositivos en escenarios como:

- Módem ↔ Router
- Router ↔ Router
- Computadora ↔ Módem
- VPN entre sitios remotos
- Redes industriales y enlaces dedicados

SLIP (Serial Line Internet Protocol)

¿Qué es SLIP?

SLIP significa Serial Line Internet Protocol. Es un protocolo muy antiguo de la capa de enlace de datos, que se usaba para enviar paquetes IP a través de conexiones seriales, como

las líneas de módem telefónico. Fue creado en 1984 y fue muy usado en sistemas UNIX. Aunque hoy está obsoleto, fue muy importante en los primeros días del Internet.

Características principales

SLIP es muy básico: no agrega cabeceras, no comprime, no corrige errores. Solo toma un paquete IP y lo manda por la línea serial. Usa un carácter especial llamado END, que vale 0xC0, para marcar el final del paquete. Además, solo sirve para IP; no funciona con otros protocolos.

Característica	**Encapsula únicamente paquetes IP.**
Encapsulación IP	Encapsula únicamente paquetes IP.
Delimitador de paquetes	Usa el carácter END (0xC0) para marcar el final de cada paquete.
Simplicidad	No incluye encabezados adicionales ni estructuras de control.
Sin detección de errores	No verifica la integridad de los datos (confía en las capas superiores).
Sin autenticación	No hay mecanismo de validación de identidad entre dispositivos.
Sin compresión	No optimiza el tamaño de los datos.
Configurfación manual	Las direcciones IP deben configurarse estáticamente.
Comunicación punto a punto	Solo funciona entre dos nodos directamente conectados.

Estructura y Funcionamiento SLIP

Estructura de una Trama de SLIP

- Esta imagen representa la estructura básica de una trama SLIP. A diferencia de otros protocolos más complejos, SLIP no utiliza una cabecera ni campos adicionales. Simplemente encapsula un paquete IP completo y agrega un carácter especial al final, llamado END, que tiene el valor hexadecimal 0xC0. Ese carácter sirve para que el receptor sepa dónde termina el paquete.”
- “Pero hay un detalle importante: si el contenido del paquete IP tiene por casualidad un byte que también sea 0xC0, SLIP lo reemplaza con una secuencia especial de escape: primero el byte ESC (que es 0xDB), seguido de 0xDC. Lo mismo ocurre si aparece un byte 0xDB en los datos; en ese caso se cambia por 0xDB 0xDD. Esto evita confusiones entre los datos reales y los delimitadores de trama.”
- “Como pueden ver en la imagen, el datagrama IP original contenía bytes C0 y DB, que fueron transformados por SLIP en esas secuencias de escape para que se transmitiera correctamente. Finalmente, se añade un END al final para cerrar la trama.”

Funcionamiento paso a paso

- **Conexión punto a punto:** Primero, se establece una conexión directa entre dos dispositivos. Generalmente se usan puertos seriales (como el COM1 o COM2) y módems para transmitir los datos a través de una línea telefónica o un cable serial. Este tipo de conexión se llama punto a punto, porque los datos viajan directamente de un dispositivo al otro, sin pasar por routers ni redes complejas.
- **Preparación del paquete IP:** Cuando uno de los equipos quiere enviar información, lo que hace es generar un paquete IP completo. Este paquete contiene los datos que deben enviarse, como si fuera una "carta" lista para enviarse por correo.
- **Encapsulamiento SLIP:** Luego, el protocolo SLIP encapsula ese paquete IP. ¿Qué significa encapsular aquí? Básicamente, solo le pone al final del paquete un carácter especial llamado END, que tiene el valor hexadecimal 0xC0. Ese carácter le dice al receptor: "Aquí termina el paquete. SLIP no agrega cabeceras, ni hace ningún otro tipo de procesamiento, lo que lo hace muy simple y rápido.
- **Transmisión por línea serial:** Después de eso, el paquete se envía byte por byte a través de la línea serial. Es como sí enviamos cada letra de una carta una por una, en orden.Como es una conexión directa y sin interrupciones, SLIP simplemente manda los datos tal cual están.
- **Manejo de caracteres especiales:**Ahora, hay un detalle importante: ¿qué pasa si el byte 0xC0 aparece dentro del contenido del paquete IP?
    - Como 0xC0 marca el fin de paquete, podría confundirse.
    - Entonces, SLIP escapa ese carácter, lo reemplaza por una secuencia especial:
    - Reemplaza 0xC0 por dos bytes: ESC (0xDB) y 0xDC
    - Esto permite que el receptor entienda que ese 0xC0 no es el fin del paquete, sino parte de los datos.
- **Recepción y reconstrucción**:Cuando el otro dispositivo recibe la transmisión, espera hasta encontrar el byte END (0xC0).
  Ese es el indicador de que el paquete terminó.
    - Luego, deshace los cambios si hubo caracteres escapados, y reconstruye el paquete IP original.
    - Finalmente, entrega el paquete al sistema operativo o a la aplicación que lo necesita.

HDLC (High-Level Data Link Control)

¿Qué es HDLC?

Es un protocolo de control de enlace de datos que funciona en la capa 2 del modelo OSI y garantiza una transmisión fiable, ordenada y libre de errores entre dispositivos conectados. Fue estandarizado por ISO/IEC 13239 y es uno de los más importantes en telecomunicaciones.

Este protocolo es orientado a bit, usando secuencias especiales llamadas banderas (01111110) para marcar el inicio y fin de cada trama. Sus funciones principales son:

- Encapsular datos en tramas.
- Gestionar la comunicación, controlando inicio, transmisión y cierre.
- Detectar y corregir errores con mecanismos como CRC.
- Controlar el flujo de datos para evitar pérdida o saturación.

Características

1. **Orientado a bit:**

   Trabaja con flujos continuos de bits, usando banderas especiales (01111110) para delimitar tramas, lo que da mayor flexibilidad y eficiencia.

2. **Estructura de tramas flexible:**

   HDLC organiza la información en tres tipos de tramas:

- I (Información): Transportan datos de usuario y control.
- S (Supervisión): Controlan flujo y confirmaciones.
- U (No numeradas): Gestionan funciones como inicio, finalización o reinicio.

3. **Detección y control de errores:**

   Usa CRC (Código de Redundancia Cíclica) para verificar la integridad de las tramas. Si hay errores, se descartan y se solicitan de nuevo.

4. **Control de flujo:**

   Emplea numeración y acuses de recibo para regular cuántas tramas se pueden enviar antes de esperar confirmación, evitando congestión o pérdida de datos.

5. **Modos de operación:**

   Funciona tanto en conexiones directas entre dos dispositivos como en configuraciones con varios nodos, donde un primario controla varios secundarios.

6. **Inserción de bits (bit stuffing):**

   Para evitar confundir datos con las banderas (01111110), se inserta un bit extra después de cinco bits consecutivos en '1' dentro de los datos.

Estructura y funcionamiento de HDLC

Estructura de una trama HDLC

La estructura define cómo están organizados los campos dentro de una trama HDLC.



- Bandera (01111110): Delimita el inicio y fin de la trama.
- Dirección: Identifica al dispositivo receptor.
- Control: Indica el tipo de trama y gestiona el control (I, S o U).
- Información: Contiene los datos reales a transmitir (solo en tramas I).
- FCS (Frame Check Sequence): Valor de verificación calculado por CRC.
- Bandera (01111110): Marca el final de la trama.

Funcionamiento

Pasos:

1. **Inicio con bandera:**
    - Se utiliza una secuencia especial `01111110` (en binario) que **delimita el comienzo de la trama**.
    - Esto permite que el receptor sepa con exactitud **cuándo empieza una nueva trama** y pueda identificarla correctamente en el flujo de bits.
2. **Campos de Dirección y Control:**
    - Dirección: Indica para quién es la trama.Es decir, identifica al dispositivo receptor dentro de la red.
    - Informa sobre el **tipo de trama** que se está enviando y gestiona aspectos de control.
        - ❖ **Trama de Información (I)** → Transporta datos del usuario.
        - ❖ **Trama de Supervisión (S)** → Gestiona control de flujo, errores y confirmaciones.
        - ❖ **Trama No Numerada (U)** → Controla funciones como inicio, parada o reinicio de la conexión.
3. **Campo de información:**
    - Contiene los datos reales que se van a transmitir(Pueden ser datos de red ,archivos ,mensajes, etc)
    - Contenido principal de la trama.
4. **FCS (Frame Check Sequence):**

    Después del campo de información, se pasa al FCS, un valor de verificación que se calcula utilizando un algoritmo llamado Cyclic Redundancy Check (CRC).

    Cómo funciona:

    1. El emisor recibe esa información y calcula el fcs con el CRC para verificar si los datos llegaron bien y completos.

2. El valor se agrega al final de la trama
3. La trama completa (datos + FCS) se envía al receptor.
4. El receptor recibe la información y el valor FCS.
5. El receptor calcula su propio FCS utilizando solo la información.
6. Ya con el valor calculado se compara los dos datos FCS (Emisor su FCS y el Receptor su FCS).

   Ya con la comparación se determinan 2 cosas importantes:

   - Si coinciden: Los datos llegan completos y sin errores. La trama se acepta.
   - Si no coinciden: Se detectó un error en la transmisión y puede solicitar una retransmisión.

5. **Cierre con otra bandera**
   - Para finalizar la trama, se utiliza nuevamente la secuencia `01111110`.
   - Esto **delimita el final de la trama**, indicando al receptor que la transmisión de esa trama específica ha terminado.

Protocolo PPP (Point-to-Point Protocol)

¿Qué es el PPP?

El Protocolo Punto a Punto (PPP) es un protocolo de capa de enlace de datos (nivel 2 del modelo OSI) diseñado para establecer una conexión directa y confiable entre dos dispositivos en una red.

Su principal objetivo es gestionar la comunicación en enlaces punto a punto, como conexiones seriales (RS-232), líneas telefónicas (dial-up), DSL (PPPoE), y redes celulares. PPP encapsula tramas de protocolos de capa de red (IP, IPv6, IPX) para su transmisión, ofreciendo mecanismos para configuración dinámica, autenticación, detección de errores y negociación de parámetros.

Características Clave

• **Soporte multiprotocolo:** PPP puede transportar datos de múltiples protocolos de capa de red (IP, IPv6, AppleTalk, IPX).

• **Autenticación integrada:** Implementa protocolos como PAP (Password Authentication Protocol) y CHAP (Challenge Handshake Authentication Protocol) para verificar la identidad de los dispositivos.

• **Control de calidad del enlace (LCP)**: Negocia parámetros como el tamaño máximo de trama (MRU), compresión de cabeceras y detección de bucles mediante "magic numbers".

• **Detección de errores:** Utiliza un Frame Check Sequence (FCS) basado en CRC para verificar la integridad de los datos (sólo detección, sin corrección).

• **Flexibilidad en medios físicos:** Funciona sobre cables de cobre, fibra óptica, radiofrecuencia e incluso conexiones virtuales (túneles).

• **Compresión y cifrado opcionales:** Mediante protocolos como CCP (Compression Control Protocol) y ECP (Encryption Control Protocol).

• **Multilink PPP:** Permite combinar múltiples enlaces físicos para aumentar el ancho de banda (RFC 1990).

Estructura y Funcionamiento

Estructura de una Trama PPP

Una trama PPP sigue un formato derivado de HDLC (High-Level Data Link Control), con los siguientes campos:

Flag	Address	Control	Protocol (2 bytes)	Datos	FCS	Flag
0x7E	0xFF	0x03	(Ej: 0x0021 para IP)	Variable	CRC	0x7E

- **Flag (1 byte):** Marca el inicio y fin de trama 0x7E
- **Address (1 byte)**:Siempre es `0xFF`, porque **no importa la dirección** (es punto a punto). Está ahí solo por compatibilidad.
- **Control (1 byte):** Siempre 0x03, Significa que **no se numeran las tramas ni se usa control de flujo**. Solo se mandan y ya.
- **Protocol (2 bytes):** Identifica el protocolo encapsulado en los datos:
    - 0xC021: LCP (control del enlace).
    - 0x8021: IPCP (configuración de IP).
    - 0x0021: Datos IP normales.
- **Datos:** Aquí van **los datos reales** que se quieren enviar, como un paquete IP, o mensajes de configuración. El tamaño depende de lo que negocien antes.
- **FCS** (2 o 4 bytes): Es un **código de verificación** (CRC-16(Parte de su diseño y rápido) o CRC-32(Es más preciso y robusto)) para detección de errores.

Funcionamiento

PPP opera en cinco fases secuenciales:

1. **Fase Dead (Inactiva):** El enlace no está activo. PPP espera un evento (como una solicitud de conexión) para iniciar la negociación.

   Después de esta fase, se pasa al Establecimiento del Enlace.

2. **Establecimiento del Enlace (LCP):** Los dispositivos intercambian paquetes LCP (Link Control Protocol) para negociar las condiciones de la conexión:

   Parámetros que negocian:

   - Tamaño máximo de trama (MRU).
   - Autenticación (si se usará PAP, CHAP o ninguna).
   - Magic Number: Número aleatorio para detectar bucles en el enlace.

Tipos de Paquetes clave LCP:

- Configure-Request: Propuesta de parámetros.
- Configure-Ack: Aceptación de parámetros.
- Configure-Nak: Rechazo de parámetros (se envía una nueva propuesta).

Cuando se termina esta negociación, se pasa a la fase de autenticación (si se acordó usarla).

3. **Autenticación (Opcional)**: Si se negoció el LCP, se ejecuta PAP o CHAP:
   - **PAP (Password Authentication Protocol)**: Envío de usuario y contraseña en texto claro. Simple, pero inseguro.
   - **CHAP (Challenge Handshake Authentication Protocol)**: Usa un reto aleatorio y un hash MD5 para validar la identidad sin enviar la contraseña directamente.

Una vez autenticados (o si no se usa autenticación), se pasa a la Negociación de Capa de Red.

4. **Negociación de Capa de Red (NCP):**
- Se negocia el protocolo de capa de red que se usará (IP, IPv6, etc.).
- IPCP (IP Control Protocol) es el más común, encargado de asignar direcciones IP dinámicas (similar a DHCP).

  Cuando se completa esta negociación, se pasa a la Transferencia de Datos.

5. **Transferencia de Datos:**
- Los datos se encapsulan en tramas PPP y se transmiten.
- **PPP detecta errores** mediante FCS, pero **la corrección depende de protocolos superiores (como TCP).**

  Cuando termina la transmisión o se desea cerrar la conexión, se pasa a la fase de Terminación.

6. **Terminación:**

- Cualquier dispositivo puede cerrar la conexión enviando un paquete LCP Terminate-Request.
- **Luego de esto, el enlace vuelve a la Fase Dead.**

Comparación de Protocolos

La siguiente tabla expone las principales características técnicas por cada protocolo.

Ventajas y desventajas

Protocolos	Ventajas	Desventajas
SLIP	• Muy ligero, simple y fácil de implementar • Bajo consumo de recursos • Compatible con sistemas UNIX y conexiones seriales.	• No incluye mecanismos de autenticación y seguridad • No soporta compresión ni cifrado • Requiere configuración manual de IP
HDLC	• Proporciona detección y corrección de errores • Soporta transmisión full-duplex • Configuración flexible (balanceada y no balanceada)	• Puede ser complejo de configurar • No es adecuado para redes multipunto • Sin compatibilidad nativa con autenticación o comprensión
PPP	• Soporta autenticación y cifrado • Asignación dinámica de IP • Compatible con múltiples tipos de redes	• Mayor consumo de recursos comparado con SLIP • Puede ser más complejo de implementar • No está optimizado para redes de alta velocidad

13

Conclusión

A lo largo del desarrollo de las redes de datos, los protocolos SLIP, HDLC y PPP marcaron etapas importantes en la evolución de los enlaces punto a punto. SLIP permitió la transmisión de paquetes IP sobre líneas seriales, aunque con limitaciones. HDLC mejoró con control de errores, flujo y tramas más estructuradas. Luego, PPP integró autenticación, encapsulación flexible y compatibilidad con múltiples protocolos, consolidándose como una solución más robusta. Aunque SLIP y HDLC han sido desplazados, sus principios siguen vigentes en estándares actuales. PPP aún se usa en entornos específicos como VPNs y enlaces satelitales. Hoy, tecnologías como Ethernet, Wi-Fi y MPLS dominan, pero los fundamentos de estos protocolos siguen siendo esenciales para comprender cómo funcionan las redes modernas.

Persona 1: Maria
A lo largo de la evolución de las redes de datos, los protocolos SLIP, HDLC y PPP han representado hitos clave en el desarrollo de mecanismos de enlace punto a punto. Cada uno fue diseñado para resolver las limitaciones del anterior, aportando mejoras en cuanto a encapsulación, control de errores, autenticación y compatibilidad con múltiples protocolos.

Persona 2:Lina
SLIP, aunque muy limitado, sentó las bases para la transmisión de paquetes IP sobre líneas seriales. Posteriormente, HDLC introdujo conceptos esenciales como la corrección de errores, control de flujo y estructuras de tramas flexibles.

Persona 3: Paula
Finalmente, PPP consolidó un protocolo más robusto, modular y adaptable a múltiples entornos físicos y lógicos. Aunque SLIP y HDLC han quedado en desuso en la mayoría de los contextos modernos, sus principios aún resuenan en los estándares actuales.

Persona 4: Alejo
PPP, por su parte, continúa vigente en aplicaciones específicas, como en redes privadas virtuales (VPN), enlaces satelitales y ciertas infraestructuras industriales, donde su capacidad de autenticación y negociación de parámetros sigue siendo valiosa.

Persona 5: Julian
En la actualidad, tecnologías como Ethernet, Wi-Fi y MPLS han desplazado estos protocolos en gran parte de las redes convencionales, ofreciendo mayor velocidad, escalabilidad y seguridad. Sin embargo, los fundamentos que dieron origen a SLIP, HDLC y PPP —como el control de enlace, la verificación de integridad y la estructuración de tramas— siguen presentes y son imprescindibles para entender el funcionamiento interno de las redes de comunicación modernas.